Architectural concept and principles

# About this document

This document is intended primary for persons working with the development. It gives readers knowledge about how the BI-solution is implemented and what rules one should follow when developing it further.

**Executive Summery**

ToDo:

# Business intelligence solution architectural overview

From an architectural view the BI-solution at PwC can be divided in to five layers

1. Visualization
2. Mart
3. Consolidated
4. Raw
5. Staging



Visualization layer

Mart layer

Consolidated layer

Raw layer

Staging layer



## Visualization

The visualization layer is where the user consumes the data. It is also named presentation layer in some cases. The data can as of today be in visualized in Qlik Sense, Excel or standard reports. Often when the data is consumed in standard reports the data is exported to Excel for further analysis.

## Mart

The mart layer is where a subset of all the data in the data warehouse is stored with the purpose for a specific data analysis. The mart can be implemented as a cube or a database. Qlik Sense is a part of both the mart layer and the visualization layer even thou Qlik in some cases reads its data from another data mart.

## Consolidated

To be able to have one truth the data has to be consolidated in one place and that is done in this layer. It is here the business rules are applied and the data from the many systems are summoned to one big data model. Data that’s not part of the general data concerning PwC e.g. external data from other companies that PwC sell data analytics, can be consolidated in a separate database.

## Raw

In some cases it is beneficial to store the data in the same format as it is gathered or delivered from the source systems. It can make the consolidation easier and increase the traceability. When this concept is applied the data is stored in the raw layer. The raw layer can contain many databases

## Staging

The staging layer is used for the initial landing area for data from the source systems. Currently all data is pulled from souse systems into staging, this may however be a subject of change to be able to handle stream analytics and real time data.

# Technical overview

From a technical point the BI solution at PwC contain eight main components, numbers within parentheses are technics that are currently under further investigation.

1. Integrations Services (SSIS): Used for extract, transform and load (ETL)
2. Database (MS SQL): Used for data warehouse and data marts
3. Analysis Services (SSAS): Data marts in form of multi-dimensional cubes
4. Qlik Sense: Used for visualization and contains a data mart in it self
5. SharePoint: Used as report portal
6. Reporting Services (SSRS): Used for standard reports
7. Master data services (MDS): Used for handling master data
8. Excel: Used mainly for browsing and reports from SSAS but is also used for handling master data. It is also common that SSRS reports are exported to Excel for further analysis.

(9.) Alteryx: The use of Alteryx for ETL is under investigation. Especially for non-automated cases as self-service and proof of concept (POC)

(10.) PowerBI: Power BI has become of interest now since SSRS support it

# Servers and databases for the new data warehouse

## Production

Primary: SE-STOSQLPWV016

Frontend: SE-STOMSCPWV003

Listener: SE-STOMSCPWV004

DW

DW

Integration server: SE-STOSQL018

Secondary: SE-STOSQLPWV015

SSAS

SE-STOSQLPWV017

## Acceptance test

SE-STOSQLPWV005

DW

Integration server: Z64TSTSQL004

SSAS

## Test

## Dev

Integration server

DW

SSAS

SE-STOSQLPWV009

Integration server

DW

SSAS

SE-STOSQLPWV008

# Documentation

The data bases are automatically documented

# General principles

The BI-solution is the single point of truth regarding all internal statistics for PwC.

The information in the data warehouse is determined by business needs and its prioritizations

The BI-Solution in not an integration platform. Application that require data should get it directly from source system. If an application needs consolidated data from the BI-solution it should preferably be done via a data mart with an optional API. This is with the exception of master data that reside in the solution, see Master Data

There should be information contract should between source systems and the BI-solution, this is as of today seldom the case.

Data from source system should preferable be gathered via lose coupled integration i.e. API, integration platform/message bus, file transfer.

Statistics about usage of the BI solution is also handled within the BI solution

GDPR require that personal data shall be removed after that the person ended its relation to PwC

Dimmensions should be as Microservices.

# De facto principles

Today most integration are made with linked servers due to the fact that the source systems doesn’t provide adequate API so a dialog with system responsible regarding changes in the application database is necessary.

Ideally contract regarding the data would be used however since direct calls are made against tables in the source system via linked server are used in many integrations this is not feasible. The same problem occurs width other external systems e.g. Salesforce where we totally rely on the API implemented by the vendor.

But since we regard our cloud applications as user friendly we trust that API are stable.

Since PwC has no centralized integration and Copernicus is a core system with limited resources the staging area in the BI-solution is used for integration

# Security

Ad-groups should be used for handling access rights. For row level security views and stored procedures are used together with a data structure called “hattar” or hats.

With the new SQL Server 2016 it will be possible to implement the row level security within the database and not rely on explicit coding in every view and stored procedure.

Some cubes has low level security today

# Design guidelines for the BI-solution

Business rules regarding the data model are implemented in the consolidated layer in the data warehouse

Data mart e.g. Qlik Sense, SSAS consume data from the data warehouse via views. The view should not contain bussiness rules.

Reports consume data from the data warehouse via stored procedures or from SSAS

Rules regarding measures i.e. calculations for reports are done in the stored procedures, simple aggregation can be performed in the report

Schemas are created from a usage perspective i.e. reports, cubs etc. More regarding information and security than technical perspective.

When an API is implemented against the BI-solution it, accessed data shall be isolated in a data mart

# Code rules

Columns in the consolidated layer shall have English names, this is of today not the case

Variables shall be in English and Camel case shall be used when the name consists of two or more words

Cubes and data marts should use the language that is specified by the owner of the system by since we move

Code shall not contain sections that are out commented. If it occurs it require a comment for this exception

To avoid unnecessary update there should be a checksum on every row, optimally compare every value, null must be handled correct

# Extract transform and load (ETL)

ETL is the part of the BI-system that connects to the source systems to fetch data and after that updates the data in the BI-solution. To distinct it from other types of data transforming the ELT needs to be scheduled and monitored to be part of the BI-system

In a wider perspective ETL is also performed in Qlik and SSAS when it read data therefore we need to monitor that as well.

On PwC for ETL we mainly use SSIS and HNO3 framework for the ETL and most data is loaded from source systems by connection to the systems database via a linked server. This will change as more applications are cloud based and the use of a linked server is not an optoion. Then c# together with SSIS is used for staging the data.

The preferred order for integration techniques are

1. System send changes
2. ETL pull changes via API
3. System provide total data
4. ETL pull total data via API????
5. ETL pull changes via linked server
6. ETL pull total data via linked server

As of today there is no integrations that delivers data to the BI-system. All integrations are implemented within the BI-solution

# Data warehouse

The data warehouse consist of three logical layers, a logical layer can consists of many database instances. The logical layers are:

1. Staging
2. Raw
3. Consolidated

## Staging

When data is read from applications and other sources it is initially written to the staging layer before it goes to the consolidated layer and alternatively to the raw layer. Currently staging is in some cases used to store the data that should be in the raw layer, this is due to that the raw is as fairly new component in the architecture.

## Raw

The purpose for the raw layer is to store the data in the way it was received from the source. The data may or may not be historized. The general principle is to historize the data but there are situations when this is not appropriate.

Currently not all data is stored in the raw layer but when a new source is added the general rule is that it should store data in the raw layer.

## Consolidated

The consolidated layer is where the data from all sources are stored so that a complete analysis will be possible. The data is divided in to dimensional data (dim) and transactional data (fact) and the tables reside in a schema named “fin”.

## Design guidelines and principles

Only data about PwC or data connected to it should reside in the fin schema. Other data e.g. data from our customers should be separated to other databases or special designated schemas.

# Data Mart

The data mart is used to enable a specific analysis of a subset of the data in the data warehouse. It is often remodeled relative the data warehouse in order to suit a specific data analysis. The data can either be pulled from the data warehouse to the mart or punched from the data warehouse to the mart. In both cases it should be done via views in the data warehouse database.

Measures are often implemented in the mart layer but more complicates business rules regarding how to interpret the data should be done in the data warehouse or in the views that is used to export the data.

Both Cubes in SSAS and qwd in Qlik are regarded as mart.

## Self Service

Data for all self-service analytics should be made available via data marts. User should not be permitted to create their own queries against the data warehouse since it is not it purpose. In general it requires good knowledge of both SQL and the data model to be able to consume the data in the data warehouse.

## Analytics Cubes – SSAS

Common dimensions shall be used in the different cubes.

Data from the data warehouse should be read via views.

## Qlik Sense

The mart part of Qlik (qvd-files) is regarded as a data marts and the same rules applies to it as for SSAS.

## Database – MS SQL

Data mart should not have any views or stored procedures that is used by the end user. If there is a risk that the data mart will give performance problem for other data bases on the server the mart should be moved to Azure or an other server.

# Report Portal and reports

SharePoint is used as the report portal for static SSRS reports. Reports gets the data by calling stored procedures in the data warehouse. There is a hat structure in the data warehouse that is used to implement row level security to ensure users only sees what there are entitled to.

The portal also have the ability to store documents as Excel but the user has to download it to be able to refresh the data. However if the soul intent with a report is to export it to Excel, the ability for Excel to read the same data from a database should be considered.

## Excel

There are two standard ways to use Excel for reporting

Either connect Excel to the SSAS cube or the database.

When using Excel for reading data direct from data marts and the data warehouse, the use of Power Query is the preferred tool.

# Master data

One fundamental difference between the data warehouse and master data is regarding integrations. As where the data warehouse is not intended to be used for integration the total opposite is the case for master data. When master data is implemented applications should be able to rely on it and use that information. As of today master data is only used in a small scale to handle some registers and mapping.

The master data for the BI-system at PwC is stored in Microsoft Master Data Services (MDS). O

## Principles

* Loading and updating the data using daily batch shall be done by a SSIS package
* Updating the data manually can be done either via Excel or the web interface
* Consuming the data shall be done via the API or via views in the database that is created in the administration portal for the MDS
* When applicable all

# Quick track

In order to be able to deliver data to a data mart and visualize it the possibility to create a quick track can be considered. When a quick track is created the preferred tools are Alteryx for the ETL and Qlik Sense for the visualization. Whether the data is intended for internal use the mart can be deployed on designated data base server and if it is intended for external use a database in Azure can be used.